STAT2_HW4

Xingyu Chen

April 08, 2022

Total Score: 21/24

1 Homework #4

See Canvas for HW #4 assignment due date.

1.1 A. Theoretical Problems

1.1.1 Problem A.1:

Let $Y_1, \ldots, Y_n \overset{i}{\sim} \text{Poisson}(\lambda_i)$. Show that, if $\eta_i = \beta_0$, then the maximum likelihood estimator of λ_i is $\widehat{\lambda}_i = \bar{Y}$, for all $i = 1, \ldots, n$.
Q1: 2/2
Answer:

likelihood function $= L(Y; \lambda)$

$$= \prod_{i=1}^{n} f(Y_i, \lambda) = \prod_{i=1}^{n} \frac{e^{-\lambda}\lambda^{Y_i}}{x_i!} = \frac{e^{-n\lambda}\lambda^{\sum_{i=1}^{n} Y_i}}{Y_1!Y_2!...Y_n!}$$

then log of Likelihood function is

$$LLF = \ln(L) = -n\lambda + \sum_{i=1}^{n} Y_i \log(\lambda) - \sum_{i=1}^{n} \log(Y_i!)$$

$$\frac{\partial}{\partial\lambda}(LLF) = -n + \frac{\sum_{i=1}^{n} Y_i}{\lambda} = 0$$

$\Rightarrow \hat{\lambda} = \frac{1}{n}\sum_{i=1}^{n} Y_i = \bar{Y} =$ sample mean.
where sample is $S = \{Y_1, Y_2, \ldots, Y_n\}$.

1.2 B. Computational Problems

1.2.1 Problem B.1

The National Institute of Diabetes and Digestive and Kidney Diseases conducted a study of 768 adult female Pima Indians living near Phoenix, AZ. The purpose of the study was the investigate factors related to diabetes.

(a) Perform simple graphical and numerical summaries of the data. Can you find any obvious irregularities in the data? If so, take appropriate steps to correct these problems.

```
# Find the data here..
pima = read.table("https://www.colorado.edu/amath/sites/default/files/attached-files/pim
#Here's a description of the data: https://rdrr.io/cran/faraway/man/pima.html
```

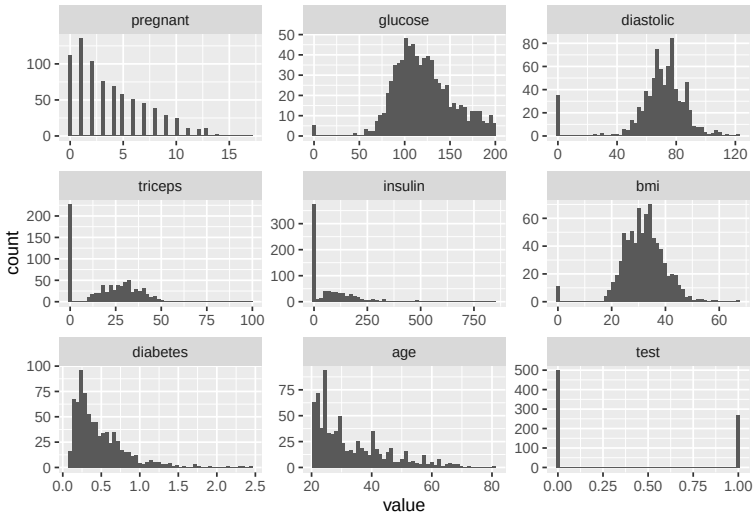
Q2: 2/2

Answer:

```
library(MASS) #Modern Applied Statistics with S
library(tidyverse)
library(reshape2)
library(caret)
summary(pima)
```

```
##     pregnant         glucose        diastolic         triceps
##  Min.   : 0.000   Min.   :  0.0   Min.   :  0.00   Min.   : 0.00
##  1st Qu.: 1.000   1st Qu.: 99.0   1st Qu.: 62.00   1st Qu.: 0.00
##  Median : 3.000   Median :117.0   Median : 72.00   Median :23.00
##  Mean   : 3.845   Mean   :120.9   Mean   : 69.11   Mean   :20.54
##  3rd Qu.: 6.000   3rd Qu.:140.2   3rd Qu.: 80.00   3rd Qu.:32.00
##  Max.   :17.000   Max.   :199.0   Max.   :122.00   Max.   :99.00
##     insulin           bmi           diabetes           age
##  Min.   :  0.0   Min.   : 0.00   Min.   :0.0780   Min.   :21.00
##  1st Qu.:  0.0   1st Qu.:27.30   1st Qu.:0.2437   1st Qu.:24.00
##  Median : 30.5   Median :32.00   Median :0.3725   Median :29.00
##  Mean   : 79.8   Mean   :31.99   Mean   :0.4719   Mean   :33.24
##  3rd Qu.:127.2   3rd Qu.:36.60   3rd Qu.:0.6262   3rd Qu.:41.00
##  Max.   :846.0   Max.   :67.10   Max.   :2.4200   Max.   :81.00
##       test
##  Min.   :0.000
##  1st Qu.:0.000
##  Median :0.000
##  Mean   :0.349
##  3rd Qu.:1.000
##  Max.   :1.000
```

```
pima_melt <- melt(pima)
ggplot(pima_melt, aes(x=value)) + geom_histogram(bins=50) + facet_wrap(~variable, scale=
```



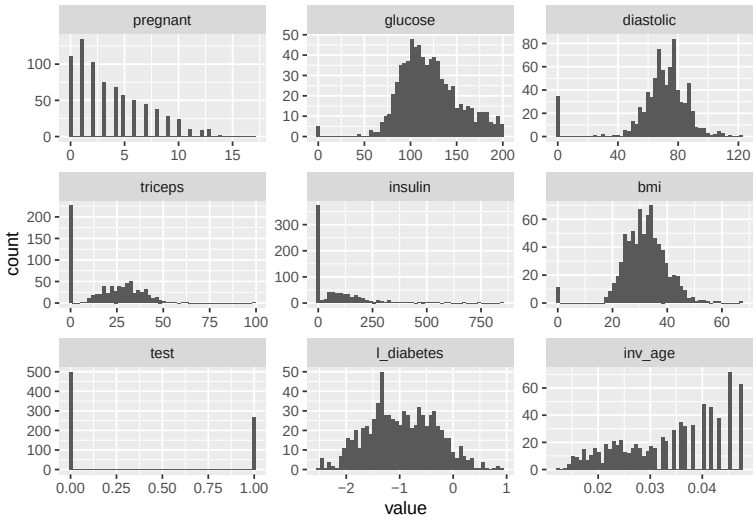

```
apply(pima, 2, BoxCoxTrans)
```

```
## $pregnant
## Box-Cox Transformation
##
## 768 data points used to estimate Lambda
##
## Input data summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   1.000   3.000   3.845   6.000  17.000
##
## Lambda could not be estimated; no transformation is applied
##
##
## $glucose
## Box-Cox Transformation
##
## 768 data points used to estimate Lambda
##
## Input data summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0    99.0   117.0   120.9   140.2   199.0
##
## Lambda could not be estimated; no transformation is applied
##
##
## $diastolic
## Box-Cox Transformation
##
## 768 data points used to estimate Lambda
##
```

```
## Input data summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   62.00   72.00   69.11   80.00  122.00
##
## Lambda could not be estimated; no transformation is applied
##
##
## $triceps
## Box-Cox Transformation
##
## 768 data points used to estimate Lambda
##
## Input data summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    0.00   23.00   20.54   32.00   99.00
##
## Lambda could not be estimated; no transformation is applied
##
##
## $insulin
## Box-Cox Transformation
##
## 768 data points used to estimate Lambda
##
## Input data summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0     0.0    30.5    79.8   127.2   846.0
##
## Lambda could not be estimated; no transformation is applied
##
##
## $bmi
## Box-Cox Transformation
##
## 768 data points used to estimate Lambda
##
## Input data summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   27.30   32.00   31.99   36.60   67.10
##
## Lambda could not be estimated; no transformation is applied
##
##
## $diabetes
## Box-Cox Transformation
```

```
##
## 768 data points used to estimate Lambda
##
## Input data summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0780  0.2437  0.3725  0.4719  0.6262  2.4200
##
## Largest/Smallest: 31
## Sample Skewness: 1.91
##
## Estimated Lambda: -0.1
## With fudge factor, Lambda = 0 will be used for transformations
##
##
## $age
## Box-Cox Transformation
##
## 768 data points used to estimate Lambda
##
## Input data summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   21.00   24.00   29.00   33.24   41.00   81.00
##
## Largest/Smallest: 3.86
## Sample Skewness: 1.13
##
## Estimated Lambda: -1.1
##
##
## $test
## Box-Cox Transformation
##
## 768 data points used to estimate Lambda
##
## Input data summary:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.349   1.000   1.000
##
## Lambda could not be estimated; no transformation is applied
```

```
pima2 <- pima[,!names(pima) %in% c("diabetes", "age")]
pima2 <- cbind(pima2, l_diabetes = log(pima$diabetes), inv_age = 1/(pima$age))
melt_pima2 <- melt(pima2)
ggplot(melt_pima2, aes(x=value)) + geom_histogram(bins=50) + facet_wrap(~variable, scale
```



there is a moderately positive high correlation between age and pregnant. There are no missing values in the dataset and hence we can proceed further. Some of the data is skewed to the right i.e. pregnant, diabetes, age. Diabetes transformation appears much more normal. The BoxCox function confirms that the log transformation of diabetes is the correct one. I will perform one more transformation as recommended above, the inverse of 'age'.

(b) Fit a model with the result of the diabetes test as the response and all the other variables as predictors. Store this model as glmod_pima. Can you tell whether this model fits the data?
Q3: 2/2
Answer:

```
glmod_pima <- step(glm(test ~ ., family="binomial", data=pima2), direction="backward")
```

```
## Start:  AIC=730.44
## test ~ pregnant + glucose + diastolic + triceps + insulin + bmi +
##     l_diabetes + inv_age
##
##               Df Deviance    AIC
## - triceps      1   712.46 728.46
## - insulin      1   713.91 729.91
## <none>             712.44 730.44
## - pregnant     1   719.36 735.36
## - diastolic    1   720.32 736.32
## - inv_age      1   722.64 738.64
## - l_diabetes   1   725.62 741.62
## - bmi          1   751.55 767.55
## - glucose      1   821.91 837.91
##
## Step:  AIC=728.46
## test ~ pregnant + glucose + diastolic + insulin + bmi + l_diabetes +
##     inv_age
```

```
##
##               Df Deviance    AIC
## - insulin      1   714.10 728.10
## <none>             712.46 728.46
## - pregnant     1   719.39 733.39
## - diastolic    1   720.45 734.45
## - inv_age      1   722.67 736.67
## - l_diabetes   1   725.87 739.87
## - bmi          1   756.49 770.49
## - glucose      1   824.11 838.11
##
## Step:  AIC=728.1
## test ~ pregnant + glucose + diastolic + bmi + l_diabetes + inv_age
##
##               Df Deviance    AIC
## <none>             714.10 728.10
## - pregnant     1   721.20 733.20
## - diastolic    1   722.47 734.47
## - inv_age      1   725.55 737.55
## - l_diabetes   1   726.53 738.53
## - bmi          1   756.76 768.76
## - glucose      1   829.31 841.31
```

```
summary(glmod_pima)
```

```
##
## Call:
## glm(formula = test ~ pregnant + glucose + diastolic + bmi + l_diabetes +
##     inv_age, family = "binomial", data = pima2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4620  -0.6993  -0.3909   0.7078   2.8100
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -5.063155   0.872816  -5.801 6.59e-09 ***
## pregnant      0.088742   0.033634   2.638 0.008328 **
## glucose       0.032757   0.003434   9.540  < 2e-16 ***
## diastolic    -0.014860   0.005190  -2.863 0.004194 **
## bmi           0.087050   0.014323   6.077 1.22e-09 ***
## l_diabetes    0.519868   0.149491   3.478 0.000506 ***
## inv_age     -42.089307  12.501627  -3.367 0.000761 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 993.48  on 767  degrees of freedom
## Residual deviance: 714.10  on 761  degrees of freedom
## AIC: 728.1
##
## Number of Fisher Scoring iterations: 5
```

(c) Using the model above, write R code to calculate the difference in the odds of testing positive for diabetes for a woman with a BMI at the first quartile compared with a woman at the third quartile, assuming all other factors are held constant. Store your answer in a variable X.
Also, give a confidence interval for this difference, stored in a variable ci .
Q4: 2/2
Answer:

```
quantile(pima2$bmi)
```

```
##   0%  25%  50%  75% 100%
##  0.0 27.3 32.0 36.6 67.1
```

```
first_quant <- subset(pima2, pima2$bmi <= 27.3)
third_quant <- subset(pima2, pima2$bmi >= 32.0 & pima2$bmi <= 36.6)
first_quant_odds <- sum(first_quant$test == 1)/(length(first_quant$test))
third_quant_odds <- sum(third_quant$test == 1)/(length(third_quant$test))
print(paste0("Odds of Testing Positive for 1st Quartile: ", first_quant_odds))
```

```
## [1] "Odds of Testing Positive for 1st Quartile: 0.103092783505155"
```

```
print(paste0("Odds of Testing Positive for 3rd Quartile: ", third_quant_odds))
```

```
## [1] "Odds of Testing Positive for 3rd Quartile: 0.449275362318841"
```

```
X <- third_quant_odds - first_quant_odds
X
```

```
## [1] 0.3461826
```

```
beta_bmi <- coefficients(glmod_pima)['bmi']
bmi_1st_quartile = 27.3
bmi_3rd_quartile = 36.6
eta_1st_quartile = bmi_1st_quartile * beta_bmi
eta_3rd_quartile = bmi_3rd_quartile * beta_bmi
diff_log_odds = eta_1st_quartile - eta_3rd_quartile
# log odds-ratio value
exp(diff_log_odds)
```

```
##       bmi
```

```
## 0.4450506
```

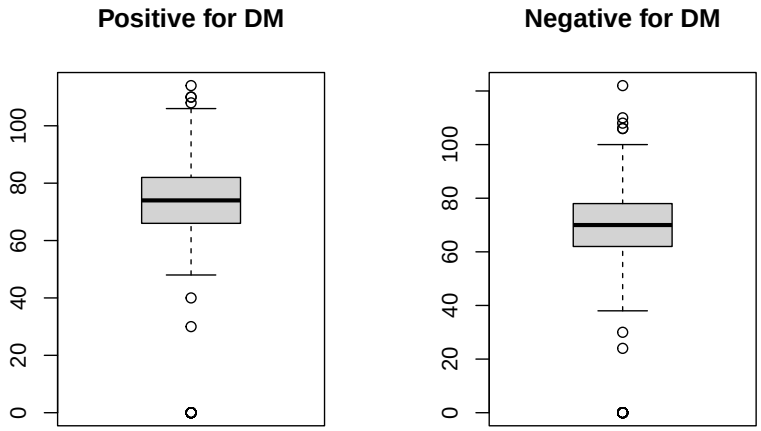
```
# calculate 95% confidence interval for bmi parameter
conf_int_bmi = confint(glmod_pima,'bmi')
# 95% confidence interval for log-odds ratio
odds_ratio = (exp(conf_int_bmi * (bmi_1st_quartile - bmi_3rd_quartile)))
# chance
ci <- odds_ratio/(1+odds_ratio)
ci
```

```
##     2.5 %    97.5 %
## 0.3647381 0.2539461
```

(d) Do women who test positive have higher diastolic blood pressures? Is the diastolic blood pressure significant in the regression model? Explain the distinction between the two questions and discuss why the answers are only apparently contradictory.
Q5: 2/2
Answer:

```
pos_dm <- subset(pima, pima$test == 1)
neg_dm <- subset(pima, pima$test == 0)

par(mfrow=c(1,2))
boxplot(pos_dm$diastolic, main="Positive for DM")
boxplot(neg_dm$diastolic, main="Negative for DM")
```




```
pos_sum <- summary(pos_dm)
print("Positive for DM, statistics for diastolic blood pressures: ")
```

```
## [1] "Positive for DM, statistics for diastolic blood pressures: "
```

```
pos_sum[,3]
```

```
##
## "Min.   :  0.00  " "1st Qu.: 66.00  " "Median : 74.00  " "Mean   : 70.82  "
##
```

```
## "3rd Qu.: 82.00  " "Max.   :114.00  "
```

```
neg_sum <- summary(neg_dm)
print("Negative for DM, statistics for diastolic blood pressures: ")
```

```
## [1] "Negative for DM, statistics for diastolic blood pressures: "
```

```
neg_sum[,3]
```

```
##
## "Min.   :  0.00  " "1st Qu.: 62.00  " "Median : 70.00  " "Mean   : 68.18  "
##
## "3rd Qu.: 78.00  " "Max.   :122.00  "
```

Yes, women who test positive have higher diastolic blood pressure: median 74 vs median 70, p<0.001, but that doesn't imply that it would be signfiicant in regression model. The data suggests they are quite similar. In the regression model, diastolic is statistically significant. These are two different meanings. Diastoic in regression means if there's significant in the way it impacts the target variable, whereas the previous question only talks about the descriptive statistics

(e) Ethical Issues in Data Collection
Read Maya Iskandarani's piece (https://researchblog.duke.edu/2016/10/24/diabetes-and-privacy-meet-big-data/)on consent and privacy concerns raised by this dataset. Summarize those concerns here. Q6: 2/2

Answer:
No researcher can realistically inform a study participant of what their medical data will be used for 40 years in the future. Generations' worth of data on the Pima tribe have been publicly accessible for over two decades. The accessibility of information as personal as blood pressure, body mass index (BMI) and number of pregnancies of Pima Native Americans, which raise privacy issue.

1.2.2 Problem B.2

The ships dataset (in the MASS package) gives the number of damage incidents and aggregate months of service for different types of ships broken down by year of construction and period of operation.

(a) The code below splits the data into a training set (80% of the data) and a test set (the remaining 20%). Use the training set to develop an appropriate regression model for the rate of incidents, using type, period, and year as predictors (HINT: is this a count model or a rate model?). Store this model in glmod_ships.
Q7: 2/2

Answer:

```
library(MASS)
data(ships)
ships = ships[ships$service != 0,]
```

```
ships$year = as.factor(ships$year)
ships$period = as.factor(ships$period)
dim(ships)
```

```
## [1] 34  5
```

```
set.seed(11)
n = floor(0.8 * nrow(ships))
index = sample(seq_len(nrow(ships)), size = n)
train = ships[index, ]
test = ships[-index, ]
head(train)
```

	type	year	period	service	incidents
40	E	75	75	542	1
28	D	65	75	192	0
18	C	60	75	552	1
19	C	65	60	781	0
5	A	70	60	1512	6
32	D	75	75	2051	4

```
dim(train)
```

```
## [1] 27  5
```

```
summary(train)
```

```
##  type  year   period     service          incidents
##  A:5   60:7   60:11   Min.   :   45.0   Min.   : 0.00
##  B:5   65:8   75:16   1st Qu.:  318.5   1st Qu.: 0.50
##  C:6   70:8           Median : 1095.0   Median : 2.00
##  D:7   75:4           Mean   : 5012.2   Mean   :10.63
##  E:4                  3rd Qu.: 2202.5   3rd Qu.:11.50
##                       Max.   :44882.0   Max.   :58.00
```

```
#Poisson distributions arive naturally when the time between events is indepedent and
glmod_ships <- glm(incidents ~ ., family=poisson, data=train)
summary(glmod_ships)
```

```
##
## Call:
## glm(formula = incidents ~ ., family = poisson, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3603  -0.5990  -0.1924   0.2763   1.9552
```

```
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.262e+00  5.349e-01  -2.359  0.01831 *
## typeB       -2.352e-01  2.966e-01  -0.793  0.42790
## typeC       -1.728e+00  4.436e-01  -3.895 9.81e-05 ***
## typeD       -8.340e-01  2.962e-01  -2.815  0.00487 **
## typeE       -4.231e-01  2.804e-01  -1.509  0.13137
## year65       2.242e+00  3.419e-01   6.560 5.40e-11 ***
## year70       2.991e+00  4.655e-01   6.426 1.31e-10 ***
## year75       2.237e+00  5.053e-01   4.426 9.58e-06 ***
## period75     8.557e-01  1.630e-01   5.249 1.53e-07 ***
## service      1.153e-04  1.567e-05   7.358 1.87e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 554.704  on 26  degrees of freedom
## Residual deviance:  27.823  on 17  degrees of freedom
## AIC: 121.53
##
## Number of Fisher Scoring iterations: 5
```

(b) Use the model that you stored in glmod_ships to calculate the mean squared prediction error (MSPE) for the test set. Store the predicted MSPE in mse_glmod_ships .

Recall from earlier assignments that the MSE can give us a sense of how well the model does at predicting new observations. The predicted mean squared error (MSE) is defined as

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

where y_i is the response in the test set, and $\hat{y}_i$ is the predicted response from glmod_ships, given the predictor values in the test set. Note that the predict.glm() function can be helpful here. Just be sure to specify the type argument (HINT: do you want $\hat{y}_i$ to be on the scale of the linear predictor η, or the mean of the response?)

Q8: 2/2

Answer:

```
mse_glmod_ships <-
  mean((test$incidents - predict.glm(glmod_ships, test, type = "response"))^2)
mse_glmod_ships
```

```
## [1] 105.8466
```

(c) Now construct a new regression model leaving out the year predictor. Store this model as glmod_ships2 . Calculate the predicted MSPE (Mean Squared Prediction Error)

for the test set using glmod_ships2 . Decide which model is better glmod_ships or glmod_ships2 - and store your answer in glmod_ships3.

Q9: 2/2

Answer:

```
glmod_ships2 <- glm(incidents ~ type + period + service, family=poisson, data=train)
summary(glmod_ships2)
```

```
##
## Call:
## glm(formula = incidents ~ type + period + service, family = poisson,
##     data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.2911  -1.6995  -0.1967   0.5213   3.8927
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.451e+00  2.020e-01   7.183 6.80e-13 ***
## typeB        1.225e+00  2.206e-01   5.551 2.84e-08 ***
## typeC       -2.047e+00  4.393e-01  -4.660 3.17e-06 ***
## typeD       -1.092e+00  2.919e-01  -3.739 0.000185 ***
## typeE       -4.915e-01  2.766e-01  -1.777 0.075511 .
## period75     7.668e-01  1.484e-01   5.166 2.39e-07 ***
## service      2.886e-05  6.393e-06   4.515 6.34e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 554.704  on 26  degrees of freedom
## Residual deviance:  94.258  on 20  degrees of freedom
## AIC: 181.96
##
## Number of Fisher Scoring iterations: 6
```

```
mse_glmod_ships2 <- mean((test$incidents - predict.glm(glmod_ships2, test, type = "respo
mse_glmod_ships2
```

```
## [1] 145.403
```

```
glmod_ships3 <- glmod_ships
```

(d) Let $\alpha = 0.05$. Conduct two χ^2 tests (using the deviance):

1. Test the adequacy of null model (store the p-value for this test in chisq_null); and

2. Test the adequacy of the glmod_ships model agaisnt the saturated model (store the p-value for this test in chisq_p).
   What conclusions should you draw from these tests?
   Q10: 1/2
   Answer:

```
nullmod <- glm(incidents ~ 1, family = poisson, data = train)
nullmod
```

```
##
## Call:  glm(formula = incidents ~ 1, family = poisson, data = train)
##
## Coefficients:
## (Intercept)
##       2.364
##
## Degrees of Freedom: 26 Total (i.e. Null);  26 Residual
## Null Deviance:       554.7
## Residual Deviance: 554.7     AIC: 630.4
```

```
chisq_null <- pchisq(554.7,26,lower.tail=FALSE)
#chisq_null <- with(anova(nullmod,glmod_ships),pchisq(Deviance,Df,lower.tail=FALSE)[2]
chisq_null
```

```
## [1] 1.597946e-100
```

```
fullmod <- glm(incidents ~ ., family = poisson, data = ships)
fullmod
```

```
##
## Call:  glm(formula = incidents ~ ., family = poisson, data = ships)
##
## Coefficients:
## (Intercept)        typeB        typeC        typeD        typeE       year65
##   1.786e-01    6.701e-01   -1.192e+00   -8.294e-01   -1.493e-01    1.087e+00
##      year70       year75     period75      service
##   1.500e+00    8.545e-01    7.284e-01    6.697e-05
##
## Degrees of Freedom: 33 Total (i.e. Null);  24 Residual
## Null Deviance:       614.5
## Residual Deviance: 70.5  AIC: 188.4
```

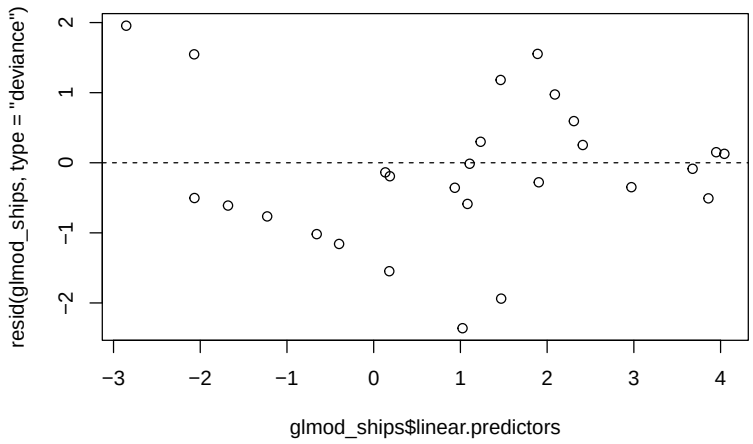
```
chisq_p <- pchisq(614.5,33,lower.tail=FALSE)
chisq_p
```

```
## [1] 2.669193e-108
```

The low p-value means we can reject the null hypohesis test and we do not need to use the saturated model.

(e) Plot the deviance residuals against the linear predictor η. Interpret this plot. Hint: The residuals function has a type parameter and "deviance" is one possible type.
Q11: 1/2
Answer:

```
plot(glmod_ships$linear.predictors, resid(glmod_ships, type='deviance'))
abline(h = 0, lty = 2)
```



polyps do not occur independently of one another, but instead may 'cluster' together. It may indicate inappropriate link function.

(f) For some GLMs (including the type in this question!), overdispersion is sometimes a problem. Overdispersion occurs when the observed (data) variance is higher than expected, if the model is correct. Explore the two models above for evidence of overdispersion.
Q12: 1/2
Answer:

```
library(AER)
#this package has a function overdispersiontest(), which conducts an overdisperion tes
#If you use it, please clearly describe the test being used, including hypotheses, tes
#and conclusions
dispersiontest(glmod_ships)
```

```
##
##  Overdispersion test
##
## data:  glmod_ships
## z = -1.1174, p-value = 0.8681
## alternative hypothesis: true dispersion is greater than 1
## sample estimates:
## dispersion
##  0.7563804
```

```
dispersiontest(fullmod)
```

```
##
##  Overdispersion test
##
## data:  fullmod
## z = 2.398, p-value = 0.008243
## alternative hypothesis: true dispersion is greater than 1
## sample estimates:
## dispersion
##   2.013165
```